Pseudocode

```
FROM pet IMPORT Pet
FROM WildAnimal IMPORT WildAnimal
FROM Treatment IMPORT Treatment
IMPORT csv

ARRAY LIST_PET
ARRAY LIST_WILDANIMAL
ARRAY LIST_TREATMENT
ARRAY ABANDON_LIST
ARRAY ABUSE_LIST
#putting csv files into variables to make them easy to call pet_file
<- "DADSA 2019-20 CWK A DATA PETS(1).csv" wildAnimal_file
<- "DADSA 2019-20 CWK A WILD ANIMALS.csv" treatment_file
<- "DADSA 2019-20 CWK A TREATMENT.csv"


#Function: load_petData
FUNCTION load_petData():     with
open(pet_file, 'r', newline='') as csvfile:
      reader <- list(csv.reader(csvfile))
#skip first row because of labels
first <- True         FOR row in reader:
         IF first:
            first <- False
ELSE:
            list_pet.append(Pet(row[0], row[1], row[2], row[3], row[4], row[5],
row[6], row[7], row[8],row[9], row[10]))
         ENDIF
      ENDFOR
ENDFUNCTION



FUNCTION load_Wildanimal():     with
open(wildAnimal_file, 'r', newline='') as csvfile:
      reader <- list(csv.reader(csvfile))
# skip first row because of labels
first <- True         FOR row in reader:
         IF first:
            first <- False
ELSE:
            list_wildAnimal.append(WildAnimal(row[0], row[1], row[2], row[3],
row[4], row[5], row[6], row[7]))
         ENDIF
      ENDFOR
ENDFUNCTION
```

```
FUNCTION load_treatment():     with
open(treatment_file, 'r', newline='') as csvfile:
reader <- list(csv.reader(csvfile))         #skip first row
because of labels
        first <- True
FOR row in reader:
            IF first:
                first <- False
ELSE:
                list_treatment.append(Treatment(row[0], row[1], row[2], row[3], row[4], row[5],
row[6], row[7]))
            ENDIF
        ENDFOR
ENDFUNCTION


FUNCTION write_new_pet():
    OUTPUT 'Please fill in details accordingly'
sanctuary_id <- input('Enter Sanctuary ID: ')
animal_type <- input('Enter Animal Type: ')     breed
<- input('Enter Animal Breed: ')     vaccinated <-
input('Enter vaccination status: ')     neutered <-
input('Enter Neutered status: ')     microchip_num <-
input('Enter Microchip Number: ')
    admission_reason <- input('What is the reason for admission: ')
arrival_date <- input('Enter arival date: ')     departure_date <-
input('Enter Departure date: ')     destination <- input('Enter
Destination: ')
    destination_address <- input('Enter Destination Address: ')

    add all inputs to the variable “row” row <- [sanctuary_id, animal_type, breed, vaccinated,
neutered, microchip_num, admission_reason,arrival_date,departure_date, destination,
destination_address]

    write to the csv
    close csv
    OUTPUT 'Pet has been added!'     restart <-
input('would you like to add another pet? y/n')     IF
restart=='y':
       Load the function write_new_pet()
ELSE:
        exit()
    ENDIF
ENDFUNCTION
```

```
FUNCTION write_new_wildAnimal():
    OUTPUT 'Please fill in details accordingly'
sanctuary_id <- input('Enter Sanctuary ID: ')
animal_type <- input('Enter Animal Type: ')     vaccinated
<- input('Enter vaccination status: ')
    admission_reason <- input('What is the reason for admission: ')
arrival_date <- input('Enter arival date: ')     departure_date <-
input('Enter Departure date: ')     destination <- input('Enter
Destination: ')
    destination_address <- input('Enter Destination Address: ')

row = [sanctuary_id, animal_type, vaccinated, admission_reason, arrival_date, departure_date,
destination, destination_address]     write to csv
        OUTPUT 'Wild animal has been added!'
        restart <- input('would you like to add another Wild animal? y/n')
IF restart = 'y':
            Call write_new_wildAnimal()
ELSE:           exitcode        ENDIF
ENDFUNCTION



FUNCTION write_new_treatment():
    OUTPUT 'Please fill in details accordingly'
sanctuary_id <- input('Enter Sanctuary ID: ')
surgery <- input('Enter surgery type: ')     surgery_date
<- input('Enter surgery date: ')     medication <-
input('Enter medication: ')
    medication_start <- input('Enter medication start date: ')
medication_finish <- input('Enter medication end date: ')
    responsible_for_abuse <- input('Name of individual that abused the animal: ')
responsible_for_abandoning <- input('name of individual that abandoned the animal: ')

    row = [sanctuary_id, surgery, surgery_date, medication, medication_start,
medication_finish, responsible_for_abuse, responsible_for_abandoning]

    Write treatment to csv file

        OUTPUT 'Treatment has been added!'
        restart <- input('Would you like to add another Treatment? y/n')
IF restart = 'y':
            CALL write_new_treatment()
ELSE:
            Exit code
        ENDIF
ENDFUNCTION



FUNCTION ready_for_adoption(pet_type):
    ARRAY adoptionStatus
    OUTPUT "this is the lisf of " + pet_type + " ready to be adopted"
for loop list_pet:
        IF animal_type = pet_type:
```

```
      IF vaccinated not empty:
IF neutered not empty
          IF microchip_num not empty:
            Add sanctuary id to adoptionStatus array
    ENDIF
      ENDIF
        ENDIF
          ENDIF
  ENDFOR
  OUTPUT adoptionStatus
ENDFUNCTION


FUNCTION BinarySearch(lsy, val):
 First =0
 Last = length of array lys – 1
 Index = -1
      WHILE first <= last and index== -1
            Mid = (first + last) // 2
            IF lys[mid].sanctuary_id == val:
                  index = mid
                  RETURN lys[index]
            ELSE
                  IF val < lys[mid].sanctuary_id :
                        last = mid - 1
            ELSE
                  first = mid + 1
            ENDIF
            ENDIF
      IF index == -1:
            RETURN index
      ENDIF
      ENDWHILE
ENDFUNCTION


FUNCTION return_pet_id_binary(return_pet):
  RETURN BinarySearch(list_pet, return_pet)
ENDFUNCTION


FUNCTION return_wild_id_binary(return_wild):
  RETURN BinarySearch(list_wildAnimal, return_wild)
ENDFUNCTION


FUNCTION return_treatment_id_binary(return_treatment):
  RETURN BinarySearch(list_treatment, return_treatment )
ENDFUNCTION
```

```
FUNCTION sort_alphabetical(array):
n <- len(array)     for i in range(n):
for j in range(0, n - i - 1):
          ENDIF
      IF array[j] > array[j+1]:            array[j],
array[j + 1] <- array[j+1], array[j]
      ENDIF
  ENDFOR
    ENDFOR
ENDFUNCTION



FUNCTION ready_to_return():
  ARRAY pet_status
  ARRAY wildAnimal_status
  ARRAY return_list
  OUTPUT “Animals ready to be returened to owner (in accending order): “
for loop in ARRAY list_pet:
    IF destination empty:
      Add pet sanctuaryid to ARRAY return_list
    ENDIF
  ENDFOR

  OUTPUT “list of wild animals ready to be returned: “
FOR loop in ARRAY list_wildAnimal:
    IF  destination is empty:
      Add wildAnimal sanctuaryid to ARRAY return_list
    ENDIF
  ENDFOR
  CALL sort_alphabetical function and pass in ARRAY return_list
  OUTPUT return_list
ENDFUNCTION
FUNCTION find_with_id(animal_id):
  isPetID = False
  FOR loop ARRAY list_pet:
IF sanctuary_id = animal_id:
      isPetID = True
      OUTPUT 'this is the pet you searched: '
      OUTPUT pet in array list_pet
    ENDIF
  ENDFOR

  IF isPetID = False:
    FOR loop ARRAY list_wildAnimal:
      IF sanctuary_id = animal_id:
        OUTPUT 'This is the wild animal you searched: '
        OUTPUT pet in array list_wildAnimal
      ENDIF
    ENDFOR
ENDIF
   FOR loop ARRAY list_treatment:
      IF sanctuary_id = animal_id:
         OUTPUT 'This is the treatment for the id you searched:
         OUTPUT pet in ARRAY list_treatment
      ENDIF
   ENDFOR
ENDFUNCTION



FUNCTION find_abandoned():
   OUTPUT 'this is the list of people who have abandoned their animals: '
FOR loop ARRAY list_treatment:
      IF responsible_for_abandoning is NOT empty and responsible_for_abandoning is NOT in
abandon_list:
         THEN ADD responsible_for_abandoning to ARRAY abandoned_list
      ENDIF
   ENDFOR
   Call sort_alphabetical function and pass in abandon_list
   OUTPUT abandon_list
ENDFUNCTION


FUNCTION find_abused():
   OUTPUT 'This is the list of people who have abusef their animals: '
FOR loop ARRAY list_treatment:
      IF responsible_for_abuse is Not empty AND responsible_for_abuse NOT in abuse_list:
ADD responsible_for_abuse to ARRAY abused_list
      ENDIF
   ENDFOR

   Call the function sort_alphabetical and pass in abuse_list
   OUTPUT abuse_list
ENDFUNCTION
FUNCTION return_pet_id(return_pet):
   foundID = False     FOR loop
Array list_pet:         IF
sanctuary_id = return_pet:
         foundID = True
         RETURN pet in ARRAY list_pet
ELSE:             foundID = False
      ENDIF
   ENDFOR

   IF foundID = False:
      OUTPUT 'pet ID was not found, please enter another ID: '
      Call function edit_menu_pet()
   ENDIF
ENDFUNCTION


FUNCTION return_Wild_id(return_wild):
   foundID = False     FOR loop
list_wildAnimal:         IF
sanctuary_id = return_wild:
```

```
          foundID ==True
          RETURN wild animal in list_wildAnimal
ELSE:            foundID ==False
      ENDIF
   ENDFOR
   IF foundID==False:
      OUTPUT 'ID not found please input another ID: '
      Call function edit_menu_wild()
   ENDIF
ENDFUNCTION

FUNCTION return_treatment_id(return_treat):
foundID=False
   FOR loop list_treatment:        IF
sanctuary_id = return_treat:
          foundID ==True
          RETURN treatment animal in list_treatment
ELSE:            foundID = False
      ENDIF
   ENDFOR
   IF foundID==False:
      OUTPUT 'ID not found, please input another ID: '
      Call function edit_menu_treat()
   ENDIF
ENDFUNCTION



FUNCTION edit_pet(pet_id_to_edit):
c = csv.reader(open(pet_file))     lines
= list(c)
   FOR loop row in lines:
      IF row[0] = pet_id_to_edit.sanctuary_id:
row[1] = pet_id_to_edit.animal_type            row[2] =
pet_id_to_edit.breed            row[3] =
pet_id_to_edit.vaccinated            row[4] =
pet_id_to_edit.neutered            row[5] =
pet_id_to_edit.microchip_num            row[6] =
pet_id_to_edit.admission_reason            row[7] =
pet_id_to_edit.arrival_date            row[8] =
pet_id_to_edit.departure_date            row[9] =
pet_id_to_edit.destination            row[10] =
pet_id_to_edit.destination_address
      ENDIF
ENDFOR
   writer = csv.writer(open(pet_file, 'w', newline=""))
writer.writerows(lines)
   OUTPUT pet_id_to_edit
ENDFUNCTION


FUNCTION edit_wild(wild_id_to_edit):
c = csv.reader(open(wildAnimal_file))
lines = list(c)
```

```
    FOR loop row in lines:
        IF row[0] = wild_id_to_edit.sanctuary_id:
            row[1] = wild_id_to_edit.animal_type
            row[2] = wild_id_to_edit.vaccinated
            row[3] = wild_id_to_edit.admission_reason
            row[4] = wild_id_to_edit.arrival_date
            row[5] = wild_id_to_edit.departure_date
            row[6] = wild_id_to_edit.destination
            row[7] = wild_id_to_edit.destination_address
        ENDIF
ENDFOR
    writer = csv.writer(open(wildAnimal_file, 'w', newline=""))
writer.writerows(lines)
    OUTPUT wild_id_to_edit
ENDFUNCTION
```

```
FUNCTION edit_treatment(treat_id_to_edit):
    c = csv.reader(open(treatment_file))
lines = list(c)
    FOR loop row in lines:
        IF row[0] = treat_id_to_edit.sanctuary_id:
            row[1] = treat_id_to_edit.surgery
            row[2] = treat_id_to_edit.surgery_date
            row[3] = treat_id_to_edit.medication
            row[4] = treat_id_to_edit.medication_start
            row[5] = treat_id_to_edit.medication_finish
            row[6] = treat_id_to_edit.responsible_for_abuse
            row[7] = treat_id_to_edit.responsible_for_abandoning
        ENDIF
ENDFOR
    writer = csv.writer(open(treatment_file, 'w', newline=""))
writer.writerows(lines)
    OUTPUT treat_id_to_edit
ENDFUNCTION
```

```
FUNCTION edit_menu_pet():
    pet_to_edit = input("Enter Pet ID: ")
    pet = call return_pet_id_binary with pet_to_edit passed in as a parameter
    OUTPUT pet
    OUTPUT "What would your like to edit? "
    OUTPUT "1. Pet type: "
    OUTPUT "2. Breed: "
    OUTPUT "3. Vaccination: "
    OUTPUT "4. Neutered: "
    OUTPUT "5. Microchip Number: "
    OUTPUT "6. Admission Reason: "
    OUTPUT "7. Arrival Date: "
    OUTPUT "8. Departure Date: "
    OUTPUT "9. Destination: "
OUTPUT "10. Destination Address: "
optionChosen = False

   WHILE not optionChosen:
        n = int(input("\n Please choose an option: "))
IF n = 1:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            animal_type = input() using .title() to capitalise first letter in input
            pet.animal_type = animal_type
            CALL edit_pet() function passing pet as a parameter
ELSEIF n = 2:             optionChosen = True
            OUTPUT "Add your new detail here: "
            breed = input() using .title() to capitalise first letter in input
pet.breed <- breed
            CALL edit_pet() function passing pet as a parameter
ELSEIF n = 3:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            vaccinated = input() using .title() to capitalise first letter in input
            pet.vaccinated = vaccinated
            call edit_pet() function passing pet as a parameter
        ELSEIF n = 4:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            neutered = input() using .title() to capitalise first letter in input
pet.neutered = neutered
            call edit_pet() function passing pet as a parameter
ELSEIF n = 5:             optionChosen = True
            OUTPUT "Add your new detail here: "
            microchip_num = input() using .title() to capitalise first letter in input
pet.microchip_num = microchip_num             call edit_pet() function
passing pet as a parameter         ELSEIF n = 6:             optionChosen =
True
            OUTPUT "Add your new detail here: "
            admission_reason = input() using .title() to capitalise first letter in input
pet.admission_reason = admission_reason             call edit_pet() function
passing pet as a parameter

        ELSEIF n = 7:
optionChosen =True
            OUTPUT "Add your new detail here: "
            arrival_date = input() using .title() to capitalise first letter in input
pet.arrival_date = arrival_date call edit_pet() function passing pet as a
parameter         ELSEIF n = 8:
            optionChosen = True
            OUTPUT "Add your new detail here: "
```

```
         departure_date = input() using .title() to capitalise first letter in input
pet.departure_date = departure_date          call edit_pet() function
passing pet as a parameter       ELSEIF n = 9:
         optionChosen <- True
         OUTPUT "Add your new detail here: "
         destination = input() using .title() to capitalise first letter in input
pet.destination = destination
         call edit_pet() function passing pet as a parameter
ELSEIF n = 10:
         optionChosen <- True
         OUTPUT "Add your new detail here: "
         destination_address = input() using .title() to capitalise first letter in input
pet.destination_address = destination_address          call edit_pet() function
passing pet as a parameter
      ENDIF
   ENDWHILE
ENDFUNCTION
```

```
FUNCTION edit_menu_wild():
   wild_to_edit = input("Enter Animal ID: ")
   wild = call return_Wild_id_binary passing wild_to_edit as a parameter
   OUTPUT wild
   OUTPUT "What would your like to edit? "
   OUTPUT "1. Animal Type: "
   OUTPUT "2. Vaccination status: "
   OUTPUT "3. Reason For Admission: "
   OUTPUT "4. Arrival Date: "
   OUTPUT "5. Departure Date: "
   OUTPUT "6. Destination: "
OUTPUT "7. Destination Address: "
optionChosen =False     while not
optionChosen:
      n = int(input("\n Please choose an option: "))
IF n = 1:
         optionChosen = True
         OUTPUT "What is the animals type: "
         type = input() using .title() to capitalise first letter in input
wild.animal_type = type
         Call function edit_wild passing wild as a parameter
ELSEIF n = 2:
         optionChosen = True
         OUTPUT "Has the animal been Vaccinated?(yes/no): "
vaccinated = input() using .title() to capitalise first letter in input
wild.vaccinated = vaccinated
```

```
          Call function edit_wild passing wild as a parameter
ELSEIF n = 3:            optionChosen = True
          OUTPUT "What is it's reason for admission: "
reason = input() using .title() to capitalise first letter in input
wild.admission_reason = reason
          Call function edit_wild passing wild as a parameter
ELSEIF n = 4:
          optionChosen = True
          OUTPUT "What is the date of arrival: "
          arrival = input() using .title() to capitalise first letter in input
wild.arrival_date = arrival
          Call function edit_wild passing wild as a parameter
ELSEIF n = 5:
          optionChosen = True
          OUTPUT "What is the date of departure: "
          departure = input() using .title() to capitalise first letter in input
          wild.departure_date = departure
          Call function edit_wild passing wild as a parameter
ELSEIF n = 6:            optionChosen = True
          OUTPUT "What is the animals final destination: "
destination = input() using .title() to capitalise first letter in input
wild.destination = destination
          Call function edit_wild passing wild as a parameter
ELSEIF n = 7:
          optionChosen = True
          OUTPUT "what is the animals destination address: "
          destination_address = input() using .title() to capitalise first letter in input
wild.destination_address = destination_address            Call function edit_wild
passing wild as a parameter
      ENDIF
   ENDWHILE
ENDFUNCTION

FUNCTION edit_menu_treat():
   treat_to_edit = input("Enter Pet ID: ")
treatment=
return_treatment_id_binary(treat_to_edit)
   OUTPUT treatment
   OUTPUT "\n"
   OUTPUT "What would your like to edit? "
   OUTPUT "1. Surgery: "
   OUTPUT "2. Surgery date: "
   OUTPUT "3. Medication: "
   OUTPUT "4. Medication start: "
   OUTPUT "5. Medication finish: "
   OUTPUT "6. Responsible for abuse: "

   OUTPUT "7. Responsible for abandoning: "
                  ENDFOR
optionChosen <- False     while
not optionChosen:
      n = int(input("\n Please choose an option: "))
IF n = 1:
          optionChosen = True
```

```
            OUTPUT "Add your new detail here: "
            animal_surgery = input() using .title() to capitalise first letter in input
treatment.surgery= animal_surgery
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 2:            optionChosen = True
            OUTPUT "Add your new detail here: "
            surgery_date = input() using .title() to capitalise first letter in input
treatment.surgery_date = surgery_date
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 3:            optionChosen = True
            OUTPUT "Add your new detail here: "
            medication = input() using .title() to capitalise first letter in input
treatment.medication = medication
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 4:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            medication_start = input() using .title() to capitalise first letter in input
treatment.medication_start = medication_start
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 5:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            medication_finish = input() using .title() to capitalise first letter in input
            treatment.medication_finish = medication_finish
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 6:
            optionChosen = True
            OUTPUT "Add your new detail here: "
            abuser = input() using .title() to capitalise first letter in input
treatment.responsible_for_abuse = abuser
            Call function edit_treatment passing treatment as a parameter
ELSEIF n = 7:            optionChosen = True
            OUTPUT "Add your new detail here: "
            abandon = input() using .title() to capitalise first letter in input
treatment.responsible_for_abandoning = abandon
            Call function edit_treatment passing treatment as a parameter
        ENDIF
    ENDWHILE
ENDFUNCTION


FUNCTION menue():
    OUTPUT 'WHAT WOULD YOU LIKE TO DO TODAY?'
    OUTPUT '1. Create a new entry  for new arrival'
    OUTPUT '2. Find the full data of an animal using its ID'
    OUTPUT '3. Produce list of identified people that have abused animals'
    OUTPUT '4. Produce list of people that have abandoned their animals'
    OUTPUT '5. Produce list of cats ready for adoption'
    OUTPUT '6. Produce list of dogs ready for adoption'
    OUTPUT '7. Produce list of animals that are ready to be returned to their owners'
OUTPUT '8. Edit stored data\n'
```

```
   selection = int(input('\nEnter a selection: '))
IF (selection = 1):
       OUTPUT 'What type of entry would you like to make?'
       OUTPUT '1. Pet'
       OUTPUT '2. wild animal'         OUTPUT
'3. Treatment'         choice = int(input('\nEnter a
selection: \n'))         IF(choice==1):
         CALL write_new_pet() function
ELSEIF(choice==2):
         CALL  write_new_wildAnimal() function
ELSEIF(choice==3):
         CALL  write_new_treatment() function
       ENDIF

 ELSEIF (selection = 2):
       animal_id = input('what is the ID of the animal you would like to look up:
')        CALL find_with_id function passing animal_id as a function     ELSEIF
(selection = 3):
       OUTPUT 'list of identified people that have abused animals'
CALL find_abused() funtion     ELSEIF (selection = 4):
       OUTPUT 'list of people that have abandoned their animals'
CALL find_abandoned() function     ELSEIF (selection = 5):
       OUTPUT 'list of cats ready for adoption'
       CALL ready_for_adoption function passing the string "Cat" as a parameter
ELSEIF (selection = 6):
       OUTPUT 'list of dogs ready for adoption'
       CALL ready_for_adoption function passing the string "Dog" as a parameter
ELSEIF (selection = 7):
       OUTPUT 'list of animals that are ready to be returned to their owners'
       Call ready_to_return() Function
   ELSEIF (selection = 8):
       OUTPUT 'Edit stored data'
       OUTPUT "1. edit pet"
                     "2. edit wild animal"
                     "3. edit treatment"

       response = int(input("?: "))

       OUTPUT response
IF response == 1:
           CALL edit_menu_pet() function

       ELSEIF response ==2:
           CALL edit_menu_wild() function

       ELSEIF response ==3:
CALL edit_menu_treat() function
ELSE:
           OUTPUT 'your response ' + response +' is an invalid selection'
w = input('would you like to restart? y/n\n')             IF w  = 'y':
              CALL menue() function
ELSE:               quit()
             ENDIF
         ENDIF
```

```
        ENDIF
ENDFUNCTION

CALL load_Wildanimal() function
CALL load_treatment() function
CALL load_petData() function
CALL menue() function
```